Efficient and Practical Audio-Visual Rendering for Games using Crossmodal Perception

David Grelaud, Nicolas Bonneel, Michael Wimmer, Manuel Asselot, George Drettakis

Efficient and Practical Audio-Visual Rendering for Games using Crossmodal Perception

David Grelaud* Nicolas Bonneel * Michael Wimmer Manuel Asselot * George Drettakis *
REVES/INRIA Sophia-Antipolis VUT REVES/INRIA Sophia-Antipolis

Abstract

Interactive applications such as computer games, are inherently audio visual, requiring high-quality rendering of complex 3D audio soundscapes and graphics environments. A frequent source of audio events is impact sounds, typically generated with physics engines. In this paper, we first present an optimization allowing efficient usage of impact sounds in a unified audio rendering pipeline, also including prerecorded sounds. We also exploit a recent result on audio-visual crossmodal perception to introduce a new level-of-detail selection algorithm, which jointly chooses the quality level of audio and graphics rendering. We have integrated these two techniques as a comprehensive crossmodal audio-visual rendering pipeline in a home-grown game engine, thus demonstrating the potential utility of our approach.

Keywords: Audio-visual rendering, crossmodal perception

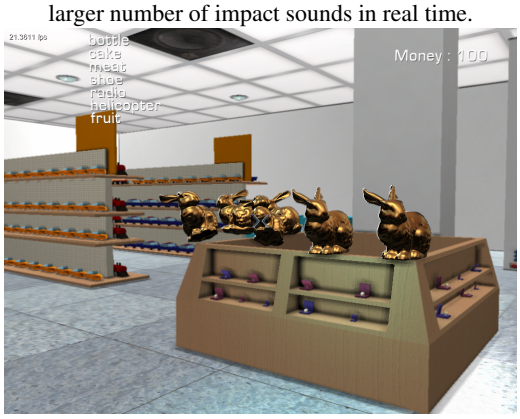
1 Introduction

Modern interactive applications, such as computer games, are inherently audio-visual. The sounds in such environments are both pre-recorded sound effects (e.g., motor sounds, ambient sounds such as wind or rain, animal sounds etc.), or can be synthesized on the fly as a result of a physical event such as an impact. We typically want to apply advanced processing to these sounds, such as spatialization. Impact sounds are an important special case, particularly in action games, since they are often responsible for the generation of a large number of sound sources, typically occurring in a short period of time. Such sounds are usually generated by physics-based interaction of objects, using physics engines (e.g., PhysX http//www.ageia.com).

Recently there have been significant advances in contact sound synthesis [van den Doel and Pai 2003; Raghuvanshi and Lin 2006; Raghuvanshi and Lin 2007], and their combined use with recorded sounds in a full scalable 3D audio pipeline [Bonneel et al. 2008]. In addition, in some of these recent works, *crossmodal perception* has been used to improve audio clustering [Moeck et al. 2007], or for scheduling of impact sounds based on perception of asynchrony [Bonneel et al. 2008].

However, the combination of impact and pre-recorded sounds involves a non-negligible overhead due to energy computation, thus mitigating the potential gains from using the techniques of audio clustering, masking and scalable processing developed in [Tsingos et al. 2004; Moeck et al. 2007]. Indeed, masking effects potentially allow the culling of a large fraction of sound sources which will not be computed. In the case of impact sounds which are costly to compute, such masking would allow a large speed-up.

Using our approach, we can render sounds for larger number of impact sounds in real time.



We use crossmodal perception to jointly select the level of detail for sound and graphics.

Figure 1: *Illustration of our results.*

While crossmodal perception has been successfully used to improve audio rendering, to our knowledge there is no previous method which uses crossmodal perception simultaneously for graphics and audio rendering, and in particular, level-of-detail selection. This could be beneficial since our experience of virtual environments is inherently bimodal, particularly if there is mutual influence of sound and visuals on perception which we can exploit algorithmically. Such mutual influence has recently been reported for our perception of materials [Bonneel et al. 2009].

In this paper we address the two issues discussed above: We present a new fast approximation to energy estimation for impact sounds, allowing us to fully exploit the benefits of the scalable processing pipeline of [Moeck et al. 2007]. We also introduce a joint audio-visual level-of-detail selection algorithm based on the aforementioned study on material perception [Bonneel et al. 2009]. We also show how to integrate high-quality "attacks" (e.g., the onset of an impact sound) into the full audio processing pipeline, and present

*e-mail: FirstName.LastName@sophia.inria.fr

a full crossmodal audio-visual processing pipeline which is suitable for games. We have integrated our results in an experimental, but quite complete, game engine [Chiu 2008], putting together all of these contributions and showing their applicability to computer games.

2 Previous Work

The work we report here is related to 3D audio rendering and crossmodal audio-visual rendering. There is a vast body of work in the perception literature on crossmodal effects, but it is beyond the scope of this paper to discuss it extensively. Please see [Spence and Driver 2004] for a review. In this section, we briefly discuss closely related previous work on 3D audio rendering and synthesis for complex environments followed by a quick overview of the most related perceptually based techniques and crossmodal audio-visual algorithms including those based on perception.

2.1 3D Audio Rendering and Synthesis for Virtual Environments

Rendering spatialized sound for 3D virtual environments has been a subject of research for many years. Techniques developed permit real-time rendering of sound reflections [Funkhouser et al. 2004] [Funkhouser et al. 1999] and [Lokki et al. 2002], mainly for pre-recorded sounds. In [Tsingos et al. 2004], sound sources are first *culled* using perceptually based auditory masking, then *clustered* to minimize computations. A scalable processing approach was added in [Moeck et al. 2007], allowing a continuous tradeoff of quality vs. computation time.

While recorded sounds are the most widespread in applications such as computer games, physically based synthesis of impact sounds [van den Doel and Pai 1998] often provides much better results. Various techniques have been developed to optimize this approach, notably recursive evaluations [van den Doel and Pai 2003] and mode-culling [Raghuvanshi and Lin 2006] which is very effective in reducing the computational overhead. Recently [Bonneel et al. 2009] use the sparsity of modes in the frequency domain to accelerate computation.

2.2 Perceptually-based methods for 3D audio and graphics

In recent years there have been many efforts to exploit perception to reduce computation for interactive virtual environments, ultimately with the goal to "render only what you can perceive". A survey of the early work in this domain can be found in [Luebke et al. 2002]. Examples of such work in graphics include use of visual differences predictors for ray-tracing acceleration (e.g., [Ramasubramanian et al. 1999; Myszkowski 1998]), or perceptually based level-of-detail (LOD) control [Luebke and Hallen 2001; Williams et al. 2003]. In audio, [Tsingos et al. 2004] uses perception to optimize masking and clustering, as discussed above.

Material perception has received quite some attention in recent years in computer graphics. Notably, [Vangorp et al. 2007] studies the effect of geometry shape and lighting on perception of material reflectance, and [Ramanarayanan et al. 2007] introduce the concept of visual equivalence, based on material properties, geometry and illumination. Our crossmodal level of detail selection approach will be based on material perception.

Crossmodal Perception Methods To our knowledge, there has been little work on exploiting crossmodal audiovisual perception to improve audio and graphics rendering. The study in [Moeck et al. 2007] indicated that it is better to have more audio clusters in the visible frustum; this was used as a new metric for audio clustering. In [Bonneel et al. 2008], tolerance to delays of audio with respect to the corresponding visual event was used to perform time-delay scheduling of impact sounds, with significant gains in performance and resulting audio quality.

A recent study [Bonneel et al. 2009] indicates a mutual influence of graphics and sound on how users perceive material. In this study, users compared approximations (spherical harmonics (SH) for BRDF rendering, and modal sound synthesis) with a high-quality solution. The users were asked to rate the *material* similarity of the approximation compared to the high-quality audio-visual rendering material. We will be developing a crossmodal algorithm to control both audio and graphics levels of detail, based on the results of this study.

3 Efficient Energy Computation for Impact Sounds

The use of a combined audio rendering pipeline for both recorded and impact sounds has a high potential for benefit, since we get significant speed benefits using masking and clustering [Tsingos et al. 2004], and we have a smooth quality/cost tradeoff using scalable processing [Moeck et al. 2007]. An initial approach integrating this pipeline with impact sounds was presented in [Bonneel et al. 2008]. However, the full potential benefit of this combined approach is hindered by the relatively high cost of the computation of impact sound energy, which is required both for masking and scalable processing.

For recorded sounds, the energy is precomputed for each sound file to avoid on-the-fly computations. In the case of impact sounds, this energy cannot be precomputed since sounds are generated on-the-fly during the simulation. Online energy computation based on the values in the current audio buffer would be costly and pointless, since the goal is to avoid computing the audio signal if it is masked by other sounds. We thus require a quick estimate of energy without actually computing the impact sound.

3.1 Energy Computations for Masking and Scalable Processing

We first briefly summarize the energy computations required to use impact sounds with the combined pipeline. Computations occur at two instants in time: at *impact* and at *each frame*. In [Bonneel et al. 2008], at each impact the total energy of each mode is efficiently computed. At each frame, the energy of the sound over the frame is then estimated using scalar products of a subset of the modes of the sound. This approximation was shown to work well for impact sound processing but still requires much computational effort per frame.

The solution we show here, together with the integration of attack processing in clustering (Sect. 5) allows the full and efficient integration of high-quality impact sounds into a practical audio processing pipeline.

3.2 An Efficient Energy Approximation for Impact Sounds

We assume that the power (energy per unit time, that we will call "instant energy") of an impact sound decreases exponentially:

$$E(t) = Ae^{-\alpha t} \tag{1}$$

Thus if we know the parameters A and α of this exponential, we can easily compute an approximation of the energy in a given frame, by analytically integrating over the desired interval. The two unknown parameters A and α satisfy two equations concerning the energy:

$$E_{Tot} = \int_0^{\infty} Ae^{-\alpha t} dt \tag{2}$$

$$E_{Part} = \int_0^{T} Ae^{-\alpha t} dt \tag{3}$$

Thus, given the total energy E_{Tot} of the sound and a partial energy E_{Part}, we are able to exactly determine parameters A and α.

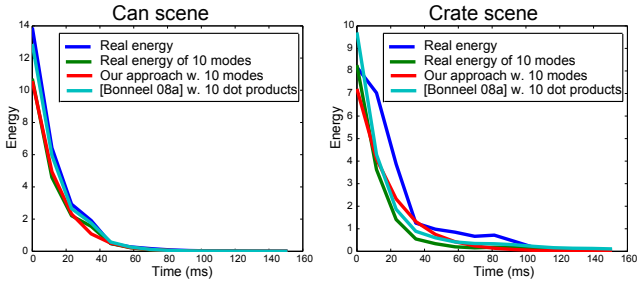


Figure 2: *Plot of the instant energies computed with our approach (red), [Bonneel et al 09] (cyan), and the reference (blue), over the length of an impact sound on two scenes. Also shown, the reference energy computed with 10 modes, simulating what our approach computes. Note that our approximation is much more efficient. Left: "Cans" sequence, right "Crates" sequence.*

These parameters are thus given by the following equations:

$$\alpha = -\frac{1}{T}\log(1 - \frac{E_{Part}}{E_{Tot}}) \tag{4}$$

$$A = \alpha\, E_{Tot} \tag{5}$$

The energy E_s for each frame is computed by integrating Eq. 1, which represents a negligible computation cost per frame:

$$E_s = -\frac{1}{\alpha}.(E(t+\Delta t) - E(t)) \tag{6}$$

However, computing these values requires the knowledge of a partial energy E_{Part} for the system to be solved. This can be achieved efficiently, also by computing scalar products. We found that the scalar product S of two modes m_1 and m_2 taken from 0 to T can be easily computed via the expression of the scalar product Q of those two modes taken from 0 to infinity:

$$\begin{aligned} S &= < m_1, m_2 > \\ &= \int_0^{T} e^{-a_1 t}\sin(\omega_1 t) e^{-a_2 t}\sin(\omega_2 t) dt \\ &= (1 - e^{-T(a_1+a_2)})Q \end{aligned} \tag{7}$$

with

$$\begin{aligned} Q &= \int_0^{\infty} e^{-a_1 t}\sin(\omega_1 t) e^{-a_2 t}\sin(\omega_2 t) dt \\ &= \frac{2(a_1+a_2)\omega_1\omega_2}{((a_1+a_2)^2+(\omega_1+\omega_2)^2)((a_1+a_2)^2+(\omega_1-\omega_2)^2)} \end{aligned} \tag{8}$$

The partial energy E_{Part} is computed by summing the pairwise scalar product of a subset of highest energy modes, similar to the computation of the total sound energy E_{Tot} previously used. A typical value for T is 10ms. This optimization process is performed only once per impact, and does not have to be repeated per frame. Also, since Q is already computed to get the total energy of the sound, the only additional per-impact cost is the computation of an exponential function. Only the simple Eq. (6) has to be computed per frame for each impact sound, which represents a negligible overhead. Our new approximation thus allows faster computation of the instant sound energy which is used for masking and budget allocation.

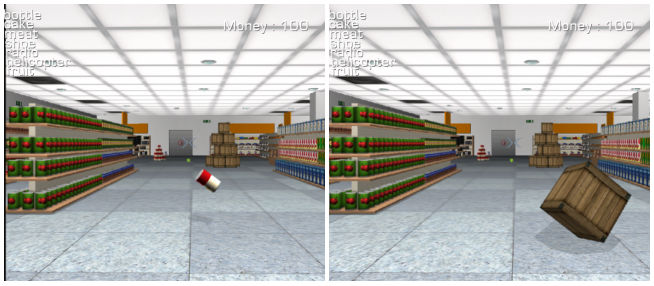


Figure 3: *Two frames from the test sequence used for the fast energy estimation evaluation (see text).*

3.3 Numerical Evaluation and Speedup

We performed a numerical evaluation of our energy estimation approach on two sequences. These scenes are respectively a can and a crate falling on the ground (see Fig. 3).

We computed the exact energy E_s of each impact sound in each frame using all the modes, and we plotted this compared to our approximation in Fig. 2, over 86 frames for both sequences. As we can see, our approximation is overall accurate, with an average L1 relative error of 24% for "Cans" and 27% for "Crates". Each can had a mesh of 415 elements, and used 113 modes; and for each crate there are 497 elements and 362 modes.

If we use the approximation of [Bonneel et al. 2008], the average cost of the energy is 0.2 ms per frame for the "Cans" sequence and 0.2 ms per frame for "Crates". In contrast, our approximation requires about 1μs and 0.8μs respectively for each sequence, corresponding to a speedup of 200 and 250 times. In addition, for sounds with large numbers of elements, which are required for sounds with fast decaying, high frequency modes, a higher number of modes is required for the approximation. Given the low cost of our approach, we can thus significantly improve the energy approximation without significant overhead in this case.

4 Crossmodal Audio-visual LOD Selection

As mentioned in Section 2, we want to develop a crossmodal audio-visual level-of-detail selection algorithm based on the perceptual study reported in [Bonneel et al. 2009]. We use the main result of that study, concerning the mutual influence of sound and graphics on how users perceive material. While this approach only applies when objects are actually sounding, i.e., during impacts, this is when a large number of sound events occur, resulting in a significant additional load to the system. As a result, there is strong potential for gain using our selection method. Indeed, since audio rendering is much cheaper than spherical harmonic (SH) lighting, we can take advantage of the crossmodal influence of audio and vi-

suals on the perceived quality, by reducing visual LOD and increasing audio LOD. According to the study, in this case the perceived similarity with a reference rendering will remain very high, despite the significantly reduced computational cost.

In the experiment, the LOD used are the number of bands of the spherical harmonics for the rendering of the BRDF using an environment map, and the number of modes used for modal sound synthesis. Also, two materials were used (gold and plastic), and two different objects were studied (bunny and dragon model). For the case of the golden bunny, we can plot the results as shown in Fig. 4. In the context of an audio-visual 3D rendering engine, we can interpret and use these results in several ways. One option would be to fix the "target quality" and then minimize the cost or computation budget used to render both the graphics and the sound. A second option is to allocate a fixed budget for both graphics and sound, and maximize the perceived quality. We have chosen the latter option, since BRDF rendering with a large number of spherical harmonic bands can be very costly; in the presence of several such objects even the joint LOD with minimum cost predicted using this model could be unacceptably high.

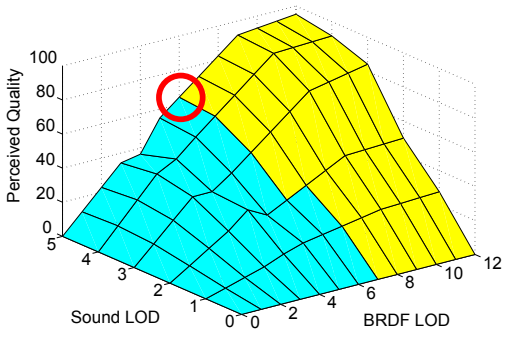


Figure 4: *Material rating depending on sound and visual LOD, and the 1.4ms threshold cost (threshold for P = 1000 shaded pixels – the size of the object on the screen). Golden bunny.*

4.1 Crossmodal Audio Visual LOD Metric

We perform an optimization step to determine the appropriate graphics and visual LODs once per frame. The constraint is determined by the actual CPU/GPU cost of rendering audio modes and shaded pixels. For graphics, this is the cost of the pixel shader used to query the SH textures and compute the dot product lighting.

The complexity of rendering audio modes is linear in the number of modes, while the cost of rendering the SH lighting is quadratic in the number of spherical harmonic bands, and linear in the number of displayed pixels. We thus use the following cost estimation:

$$C_{AV} = C_M\, M + C_S\, S^2\, P, \tag{9}$$

where M is the number of modes, S is the number of spherical harmonic bands, P is the number of shaded pixels, and C_M and C_S are the costs of rendering respectively one audio mode and one band of one SH-shaded pixel. The values for C_M and C_S can be measured once for every hardware setup. In our case $C_M = 5.23\mu s/mode$, $C_S = 0.0368\mu s$ per SH coefficient and per pixel. Note that rendering S bands requires computing S^2 such coefficients. These values were measured with a GeForce8800 GTX, and a 2.3GHz Intel Xeon CPU. To efficiently determine the number of shaded pixels, we use an early-depth pass, deferred shading and occlusion queries. The above expression shows the quadratic increase in SH cost and the linear cost in the number of modes.

The target cost C_T is a user-defined parameter typically depending on the hardware setup. We share this parameter across all objects, i.e., if we have N objects, the budget of C_T/N ms is assigned to each object in the scene. The audio-visual perceived quality is determined by an interpolation of the tabulated values given in [Bonneel et al. 2009], Fig.7. We evaluate the cost function C_{AV} for each combination of 13 SH bands (0 to 12) and 6 mode budgets given in [Bonneel et al. 2009]. We choose the combination which results in the highest "perceived quality" as determined by the values reported in the study. As an example, for a target budget C_T/N of 1.4ms, used to render an object occupying 1000 pixels on the screen, we can visualize the operation LOD choice operation in Fig. 4. In this example, we will choose 6 SH bands and level 5 for the modes. We perform this operation for each object, and select the number of SH bands and the number of modes resulting in the highest quality, while respecting the target cost.

With this approach we determine a budget for impact sounds and spherical harmonics. The audio LOD is used for the current audio frame, and the spherical harmonic LOD is used in the next visual frame. Visual blending is performed in the pixel shader by progressively adding terms in the dot product (without splitting an SH band) during 0.5ms.

Evidently this LOD selection is valid only when the objects are actually making a sound, in this case when impacts occur. When the objects are not sounding, we use a fixed LOD for graphics, i.e., 5 spherical harmonic bands. The choice of 5 bands is also based on the results presented in [Bonneel et al. 2009], since we can see that perceived quality with this approximation is not much different from the perceived quality obtained with the highest visual LOD, for all sound LODs. The crossmodal LOD selection mechanism is applied at the first audio frame of an impact, and maintained for a short period of time (typically 7 seconds).

5 A General Crossmodal Audiovisual Pipeline

The fast energy estimation and the crossmodal audiovisual LOD selection pave the way for us to introduce a general crossmodal rendering pipeline, in which we use crossmodal perception for combined graphics and sound resource management.

Attack processing with Clusters In [Bonneel et al. 2009], special processing is performed to provide high-quality rendering of the "attacks" or onset of impact sounds. This involves a specially designed windowing operation in the first frame of each attack. This approach was however not integrated into the full pipeline.

The inclusion of high-quality attack processing in the pipeline is important to allow high-quality audio rendering. In each cluster, we have a set of recorded sounds and a set of impact sounds. Using the improved attack processing of [Bonneel et al. 2008], we need to have one frequency domain buffer for attacks, and one for both recorded and impact sounds. Budget allocation for scalable processing is performed only in the latter buffer. Windowing operations are performed separately in each buffer, and an inverse FFT is computed for each *impact sound* buffer in each cluster. Recorded sound buffers are processed together, resulting in a single inverse FFT. Attack buffers and the recorded/impact sound buffers are then correctly windowed in the time domain to produce the final sound.

The additional overhead of this approach is thus one inverse FFT per audio channel.

General Crossmodal AV Pipeline We have implemented a complete perceptually based audio-visual rendering pipeline. We have also included the crossmodal audio clustering metric [Moeck et al. 2007], which assigns more audio clusters to the viewing frustum, the crossmodal scheduling approach of [Bonneel et al. 2008] which significantly improves performance, and the audio-visual LOD selection introduced here.

At each frame, both for audio and graphics, we evaluate the parameters used for each of these crossmodal algorithms, and set the appropriate values for clustering, impact sound scheduling and AV LOD selection.

6 Results

We have implemented the general AV pipeline into a home-grown "production level" game engine developed in our institutions [Chiu 2008]. All the examples shown in this paper and the accompanying video are taken from this engine.

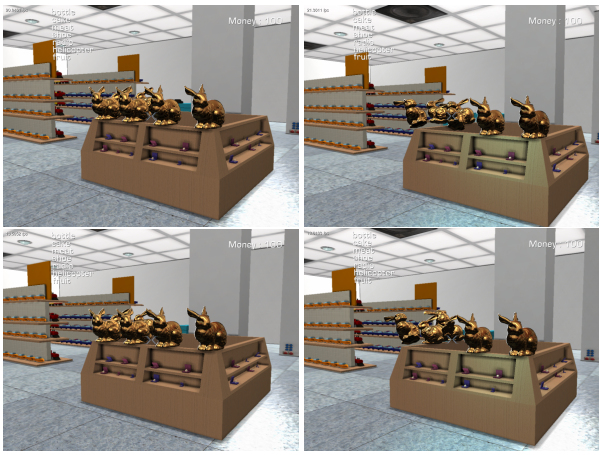


Figure 5: *(a)-(b) Two frames of our test sequence running at 26 fps using crossmodal LOD manager. (c)-(d) Two similar frames using highest LOD running at 18 fps.*

Energy approximation and attack integration The combination of the integration of attacks in the pipeline and the energy approximation now make it possible to have more impact sounds with higher quality in complex game-like settings. In the accompanying video, we show an example game play, in which we have 7 recorded sounds (toys, stereos, door, cashier sounds etc.) and a large number of impact sounds, with a peak of 712 impacts just after the piles of boxes and cans are knocked over.

We first show the sequence using the method of [Bonneel et al. 2008]. The cost of the previous energy estimation method results in low quality audio processing, which is clearly audible. If we compare to a similar sequence (we do not have exact path replay in our implementation), using our energy approximation we see that the quality achievable by the scalable processing is significantly improved.

We next show the same sequence first without attack processing in the clusters and using the attacks, both with the energy approximation. As we can hear, the quality of the impact sounds is audibly better using the attack processing approach; thanks to the integration with clustering, the overhead is minimal. Rendering this scene without clustering, given the number of sounds, would be impossible in real time.

We provide a detailed breakdown of the costs of the different stages for the above sequence, using both the energy approximation and the integration of attacks in clusters (see Table 1). In the scenes we tested, we achieve masking of around 60% of the contact sounds.

Stage	Time (ms)
Masking	$3.339E-3$
Clustering	$4.652E-3$
Impact sound synthesis	$2.759E-3$
Energy	$0.824E-3$
Scalable mixing	$1.260E-2$

Table 1: *Time in ms for each stage of the unified pipeline*

Crossmodal Material LOD We have implemented our crossmodal LOD selection approach in the same game engine [Chiu 2008]. In the accompanying video we show a sequence containing two piles of golden bunnies which are accidentally knocked over. In Fig. 5 we illustrate the choice of visual LODs for two frames of the sequence. We used a budget of 100 ms shared by all SH-shaded objects. Note that this budget will be shared by the GPU for SH lighting computations and the CPU for modes generation, both running in parallel. This budget does not reflect the exact running time of the simulation but is used as an indication.

For comparison, we computed an appropriate fixed audio-visual level of detail, corresponding to the perceived similarity to the reference rendering, as predicted by the perceptual study. For example, if the average choice of LOD in the sequence represents 90% similarity to the reference, we will choose 90% of the spherical harmonic bands and the modes used for the reference.

On average across the entire sequence, the model of [Bonneel et al. 2009] predicts that we have a quality level of 90.6%, i.e., the rating of perceived similarity to the high quality rendering. The computation in this sequence is 44% faster; on average we achieved 26 fps using our approach and 18 fps using the equivalent quality with fixed LOD.

Full crossmodal approach There are three main differences with [Bonneel et al. 2008]: we integrate high-quality attack processing with clustering, the fast energy approximation, and the crossmodal LOD. We integrated these effects with the previous crossmodal metrics (clustering and scheduling) to provide a unified crossmodal pipeline.

The final sequence of the video shows all the elements working together. We selectively turn on and off some of the features, illustrating the various benefits obtained.

7 Discussion and Conclusion

We have presented a complete perceptually inspired crossmodal audio-visual rendering pipeline for games. We introduced an approximate energy estimation method for contact sounds, and showed how to integrate high-quality attacks with clustering. We also presented a crossmodal LOD selection approach, which is to our knowledge the first algorithm which jointly chooses audio and graphics levels of detail, based on crossmodal perception. We believe that the integrated method presented here offers a practical and useful pipeline for audio visual rendering, which can be useful for games or other similar applications.

Our crossmodal LOD selection approach uses spherical harmonic rendering, since we base our algorithm on the study presented in [Bonneel et al. 2009] which used this approach. We expect that similar results can be obtained with other approaches for environment map rendering (e.g., zonal harmonics [Sloan et al. 2005] or sampling techniques [Agarwal et al. 2003]).

The potential for computation gains can be significant, especially during events which result in large numbers of impacts. Nonetheless, the work reported here is only a first step. In particular, we need to determine how well the results of [Bonneel et al. 2009] generalize to other kinds of materials and objects, and to determine how perceived quality ratings are affected by the more realistic game settings shown here. Our intuition is that perceived quality should actually be higher in the realistic setting compared to the experimental condition. We also expect that a generic material model, based for example on some general material categories, could be applied in a more general setting. One way to do this would be to sample a perceptually uniform material space such as [Pellacini et al. 2000] and interpolate the results for any material. Also, the environment map used for this game setting was different from the one used in the experiment in [Bonneel et al. 2009], since this new environment is an indoors environment, contrary to the previous experiment. It has been shown that the environment lighting influences the perception of materials ([Fleming et al. 2003]), and a further BRDF prediction could include this parameter. In our case, an indoor environment makes the golden bunnies appear a bit more diffuse than with a higher frequency environment.

Acknowledgments

This research was funded by the EU FET project CROSSMOD (014891-2 http://www.crossmod.org). We thank Autodesk for the donation of Maya, Fernanda Andrade Cabral for modeling and the anonymous reviewers for their helpful suggestions.

References

AGARWAL, S., RAMAMOORTHI, R., BELONGIE, S., AND JENSEN, H. W. 2003. Structured importance sampling of environment maps. In *ACM Trans. on Graphics (Proc. of SIGGRAPH)*, vol. 22, 605–612.

BONNEEL, N., DRETTAKIS, G., TSINGOS, N., VIAUD-DELMON, I., AND JAMES, D. 2008. Fast modal sounds with scalable frequency-domain synthesis. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 27*, 3 (August), 1–9.

BONNEEL, N., SUIED, C., VIAUD-DELMON, I., AND DRETTAKIS, G. 2009. Bimodal perception of audio-visual material properties for virtual environments. *ACM Transactions on Applied Perception (Accepted with minor revisions)*.

CHIU, D.-F. 2008. *Penta G - A Game Engine for Real-Time Rendering Research*. Master's thesis, Institute of Computer Graphics and Algorithms, Vienna University of Technology.

FLEMING, R. W., DROR, R. O., AND ADELSON, E. H. 2003. Real-world illumination and the perception of surface reflectance properties. *Journal of Vision 3*, 5 (July), 347–368.

FUNKHOUSER, T., MIN, P., AND CARLBOM, I. 1999. Real-time acoustic modeling for distributed virtual environments. In *Proc. of ACM SIGGRAPH 99*, 365–374.

FUNKHOUSER, T., TSINGOS, N., CARLBOM, I., ELKO, G., SONDHI, M., WEST, J., PINGALI, G., MIN, P., AND NGAN, A. 2004. A beam tracing method for interactive architectural acoustics. *The Journal of the Acoustical Society of America (JASA), 2003. 115*, 2 (February), 739–756.

LOKKI, T., SAVIOJA, L., VÄÄNÄNEN, R., HUOPANIEMI, J., AND TAKALA, T. 2002. Creating interactive virtual auditory environments. *IEEE Comput. Graph. Appl. 22*, 4, 49–57.

LUEBKE, D., AND HALLEN, B. 2001. Perceptually driven simplification for interactive rendering. In *Proc. of EG Workshop on Rendering 2001*, 223–234.

LUEBKE, D., WATSON, B., COHEN, J. D., REDDY, M., AND VARSHNEY, A. 2002. *Level of Detail for 3D Graphics*. Elsevier Science Inc., New York, NY, USA.

MOECK, T., BONNEEL, N., TSINGOS, N., DRETTAKIS, G., VIAUD-DELMON, I., AND ALLOZA, D. 2007. Progressive perceptual audio rendering of complex scenes. In *ACM SIGGRAPH Symp. on Interactive 3D Graphics and Games (I3D)*, 189–196.

MYSZKOWSKI, K. 1998. The Visible Differences Predictor: applications to global illumination problems. In *Proc. of EG Workshop on Rendering 1998*, 223–236.

PELLACINI, F., FERWERDA, J. A., AND GREENBERG, D. P. 2000. Toward a psychophysically-based light reflection model for image synthesis. In *Proc. of ACM SIGGRAPH 00*, 55–64.

RAGHUVANSHI, N., AND LIN, M. C. 2006. Interactive sound synthesis for large scale environments. In *ACM SIGGRAPH Symp. on Interactive 3D Graphics and Games (I3D)*, 101–108.

RAGHUVANSHI, N., AND LIN, M. C. 2007. Physically based sound synthesis for large-scale virtual environments. *IEEE Comput. Graph. Appl. 27*, 1, 14–18.

RAMANARAYANAN, G., FERWERDA, J., WALTER, B., AND BALA, K. 2007. Visual equivalence: Towards a new standard for image fidelity. *ACM Trans. on Graphics (Proc. of SIGGRAPH)* (August), 76.

RAMASUBRAMANIAN, M., PATTANAIK, S. N., AND GREENBERG, D. P. 1999. A perceptually based physical error metric for realistic image synthesis. In *Proc. of ACM SIGGRAPH 99*, 73–82.

SLOAN, P.-P., LUNA, B., AND SNYDER, J. 2005. Local, deformable precomputed radiance transfer. In *ACM Trans. on Graphics (Proc. of SIGGRAPH)*, vol. 24, 1216–1224.

SPENCE, C., AND DRIVER, J. 2004. *Crossmodal Space and Crossmodal Attention*. Oxford University Press, USA, June.

TSINGOS, N., GALLO, E., AND DRETTAKIS, G. 2004. Perceptual audio rendering of complex virtual environments. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 23*, 3 (July), 249–258.

VAN DEN DOEL, K., AND PAI, D. K. 1998. The sounds of physical shapes. *Presence 7*, 4, 382–395.

VAN DEN DOEL, K., AND PAI, D. K. 2003. Modal synthesis for vibrating objects. *Audio Anecdotes*.

VANGORP, P., LAURIJSSEN, J., AND DUTRÉ, P. 2007. The influence of shape on the perception of material reflectance. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 26*, 3 (August), 77.

WILLIAMS, N., LUEBKE, D., COHEN, J. D., KELLEY, M., AND SCHUBERT, B. 2003. Perceptually guided simplification of lit, textured meshes. In *ACM SIGGRAPH Symp. on Interactive 3D Graphics and Games (I3D)*, 113–121.